# *ng8-o2-auth-fb* Firebase authentication service library (for Angular8)

*ng8-o2-auth-fb* is a service library for using Firebase's authentication service, which is developed by Angular8.

*Video Explanation*
<https: //>

*Sample Project all sources*
https://github.com/Ohtsu/ng8-o2-auth-fb-project (https://github.com/Ohtsu/ng8-o2-auth-fb-project)

## Overview

- *ng8-o2-auth-fb* is a component that is independent of the user interface. Therefore, you can support both Angular Material and Bootstrap.

- Supported certificates are Google, Twitter, Facebook, GitHub, as well as login and password.

- User registration information is registered in the Firebase Firestore database.

- Responds to forgot password issue.

- Depending on environment variables, you can set pages such as transition destination after login.

- When you select the debug mode, you can get login status etc in the console.

## Required environment

- Node.js
- TypeScript3
- firebase
- @angular/router
- @angular/fire

## Operating environment version of the library

- @angular/common: 8.0.0
- @angular/core: 8.0.0
- @angular/router: 8.0.0
- @angular/fire: 5.1.3
- firebase: 6.1.0

## Install

In order to install this library, install with *npm* as follows.

```
$ npm i ng8-o2-auth-fb
```

## Registered methods

The following methods are registered in this library.

| Method | function | argument | return value |
|--------|----------|----------|--------------|
| isAuthenticated() | get authentication status | none | true: authenticated false: not authenticated |
| getLoginState() | Get current user information | none | user data |
| getCurrentUser() | Get current user information | none | user data |
| setUserData(user) | Save current user information to Firebase | user data | registration result |
| loginEmail(email: string, password: string) | login | email, password | return setUserData (user) |
| signUp(email, password) | user registration | email, password | return setUserData (user) |
| loginGoogle() | Login with Google Account | None | return setUserData(user) |
| loginFacebook() | Login with Facebook account | None | return setUserData(user) |
| loginTwitter() | login with Twitter account | none | return setUserData (user) |
| loginGithub() | Login with GitHub account | None | return setUserData(user) |
| forgotPassword(passwordResetEmail) | Reissue password | email | None |

The above user data has the following structure. It is saved in the *user* directory of the Firebase Firestore database in this format. Please be aware that this function can not be used unless Firestore database is made available.

```
export interface User {
    uid: number;
    email: string;
    displayName: string;
    emailVerified: boolean;
    photoURL ?: string;
    providerId ?: string;
    isAnonymous ?: boolean;
}
```

## Configurable environment variables

The environment variables that can be set in this library are as follows.
This is specified in the `environments/environment.ts` file of Angular's project.

| Variable name | Meaning |
|---|---|
| debugMode | true: Output authentication information to console false: Do not output |
| redirectPath.afterLoginPath | Path after login |
| redirectPath.signInPath | login path |
| redirectPath.registerUserPath | User registration path |
| redirectPath.dashboardPath | Dashboard page path (normal top page) |
| redirectPath.forgotPasswordPath | Path if you forget your password |
| message.forgotPasswordEmailSent | Message when password is reissued |
| message.logout | Message when logging out |

Please note that this environment variable may increase from time to time due to version upgrade.

```
o2AuthService: {
  debugMode: true,
  redirectPath: {
    afterLoginPath: '/',
    signInPath: 'sign-in',
    registerUserPath: 'register-user',
    dashboardPath: 'dashboard',
    forgotPasswordPath: 'forgot-password',
    verifyEmailPath: 'verify-email-address'
  },
  message: {
    sendChangeEmail: 'Sent email',
    logout: 'Logout',
    forgotPasswordEmailSent: 'Password reset email sent, check your inbox',
    resetPasswordEmailSent: 'Password update email sent'
  }
},
```

## Sample Project

*Ng8O2AuthFbProject* is a sample project for making and using Firebase Authentication Library ( *ng8-o2-auth-fb* ).

The sample project uses Angular Material, but this service library can also be used in Bootstrap.

*This full source code,*
https://github.com/Ohtsu/ng8-o2-auth-fb-project (https://github.com/Ohtsu/ng8-o2-auth-fb-project)

*Video Explanation (Japanese),*
https://youtu.be/ (https://youtu.be/)

*Video Explanation (English),*
https://youtu.be/ (https://youtu.be/)

<img src="https://raw.githubusercontent.com/Ohtsu/images/master/ng7/FontAweSomeInAngular8_02.png" width= "640" >

### Sample Project Prerequisite

- node.js
- Typescript3
- Angular8
- Angualr Material
- Firebase
- Angular/fire
- ng8-o2-auth-fb

### Sample Project Installation

First move to the appropriate directory and download the project as follows.

```
$ git clone https://github.com/Ohtsu/ng8-o2-auth-fb-project.git
```

Then move to the project directory.

```
$ cd ng8-o2-auth-firebase-project
```

To install this project, run simply:

```
$ npm install
```

### Set environment variables

Before starting the server, you need to create an account on Firebase and set that information as an environment variable.

We need to make Firestore database available as well.

Environment variables are set in the environments/environment.ts file.

```
export const environment = {
  production: false,
  firebase: {
    apiKey: '-- You need to set this param--',
    authDomain: '-- You need to set this param--',
    databaseURL: '-- You need to set this param--',
    projectId: '-- You need to set this param--',
    storageBucket: '-- You need to set this param--',
    messagingSenderId: '-- You need to set this param--'
  },

  o2AuthService: {
    debugMode: true,
    redirectPath: {
      afterLoginPath: '/',
      signInPath: 'sign-in',
      registerUserPath: 'register-user',
      dashboardPath: 'dashboard',
      forgotPasswordPath: 'forgot-password',
      verifyEmailPath: 'verify-email-address'
    },
    message: {
      sendChangeEmail: 'Sent email',
      logout: 'Logout',
      forgotPasswordEmailSent: 'Password reset email sent, check your inbox',
      resetPasswordEmailSent: 'Password update email sent'
    }
  }
};
```

**Start local server**

Start local server as follows.

```
$ ng s -o
```

You will find default dashboard page in your browser.

- *First Page*

<img src="https://raw.githubusercontent.com/Ohtsu/images/master/ng8-o2-auth-firebase/ng8-o2-auth-firebase_00.png" width= "640" >

- *Login Page*

When you click the login button, it is displayed as follows.
<img src="https://raw.githubusercontent.com/Ohtsu/images/master/ng8-o2-auth-firebase/ng8-o2-auth-firebase_01.png" width= "640" >

- *Check Email format*

When you enter the email, the incorrect email address will be checked.

<img src="https://raw.githubusercontent.com/Ohtsu/images/master/ng8-o2-auth-firebase/ng8-o2-auth-firebase_02.png" width= "200" >

- *Check Password length*

When you click the register button, it is displayed as follows. And password length will be checked.

<img src="https://raw.githubusercontent.com/Ohtsu/images/master/ng8-o2-auth-firebase/ng8-o2-auth-firebase_03.png" width= "640" >

- *Check Password matching*

Matching between password and confirm password will be checked.

<img src="https://raw.githubusercontent.com/Ohtsu/images/master/ng8-o2-auth-firebase/ng8-o2-auth-firebase_04.png" width= "640" >

- *Debug Mode*

Setting debugMode of the environment variable to true makes it possible to obtain information in console as follows.

```
o2AuthService: {
    debugMode: true,
```

<img src="https://raw.githubusercontent.com/Ohtsu/images/master/ng8-o2-auth-firebase/ng8-o2-auth-firebase_05.png" width= "640" >

**Version**

- ng8-o2-auth-fb-project : 0.3
- ng8-o2-auth-fb : 0.3
- Angular/cli : 8.0.0
- TypeScript : 3.1.6
- firebase : 5.8.2
- @angular/fire : 5.1.1
- @angular/material : 7.3.1
- @fortawesome/fontawesome-free : 5.7.1
- hammerjs : 2.0.8
- node.js : 10.12.0

**Change Log**

- 2019.5.31 version 0.3 uploaded

## Reference

- "Angular + Firebase Progressive Web App Starter",
  https://github.com/codediodeio/angular-firestarter/edit/master/src/app/core/auth.service.ts (https://github.com/codediodeio/angular-firestarter/edit/master/src/app/core/auth.service.ts)

- "Full Angular 7 Firebase Authentication System",
  https://www.positronx.io/full-angular-7-firebase-authentication-system/?fbclid=IwAR1pXk_hPy_77gigtmpKXIhAPuEY_0MVuX9ssxSJsWUgA71m6w4LQ1XimPI#.XDk-Jxx4HcA.facebook (https://www.positronx.io/full-angular-7-firebase-authentication-system/?fbclid=IwAR1pXk_hPy_77gigtmpKXIhAPuEY_0MVuX9ssxSJsWUgA71m6w4LQ1XimPI#.XDk-Jxx4HcA.facebook)

- "Angular 7 - Reactive Forms Validation Example",
  http://jasonwatmore.com/post/2018/11/07/angular-7-reactive-forms-validation-example (http://jasonwatmore.com/post/2018/11/07/angular-7-reactive-forms-validation-example)

- "Angular Material Form Validation, Input, Datepicker and Modal",
  https://code-maze.com/angular-material-form-validation/ (https://code-maze.com/angular-material-form-validation/)

- "[Angular エラー対策] Expected validator to return Promise or Observable",
  https://qiita.com/seteen/items/50999fc33a27bd07e520 (https://qiita.com/seteen/items/50999fc33a27bd07e520)

- "Angular 6, Angular 7 Custom Library: Step-by-step guide",
  https://www.udemy.com/angular5-custom-library-the-definitive-step-by-step-guide/ (https://www.udemy.com/angular5-custom-library-the-definitive-step-by-step-guide/)

- "Angular 6, Angular 7用 カスタムライブラリの作成: 完全ステップ・バイ・ステップ・ガイド",
  https://www.udemy.com/angular5-l/ (https://www.udemy.com/angular5-l/)

## Copyright

copyright 2018 by Shuichi Ohtsu (DigiPub Japan)

## License

MIT © Shuichi Ohtsu (ohtsu@digipub-net.com)